COMP 576 - Fall 2017
Assignment 1

October 3, 2017

Backpropagation in a Simple Neural Network

1a) Dataset

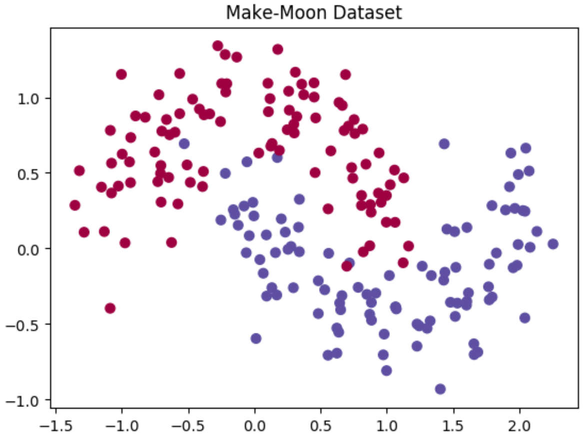

Figure 1: Make Moon Dataset

1b) Derivatives of Activation Functions

Sigmoid:

$$f(x) = \frac{1}{1+e^{-x}} = (1+e^{-x})^{-1}$$

$$\frac{d(1+e^{-x})^{-1}}{dx} = (1+e^{-x})^{-2}(e^{-x})$$

$$= \frac{e^{-x}}{(1+e^{-x})^2} = \frac{1}{1+e^{-x}}\frac{e^{-x}}{1+e^{-x}}$$

$$= \frac{1}{1+e^{-x}}\frac{(1+e^{-x})-1}{1+e^{-x}}$$

$$= \frac{1}{1+e^{-x}}\left(1-\frac{1}{1+e^{-x}}\right)$$

$$\boldsymbol{\frac{d(1+e^{-x})^{-1}}{dx} = f(x)(1-f(x))}$$

Tanh:

$$f(x) = \tanh x = \frac{\sinh x}{\cosh x}$$

$$\frac{d\left(\frac{\sinh x}{\cosh x}\right)}{dx} = \frac{\cosh x \cosh x - \sinh x \sinh x}{\cosh^2 x}$$

$$= \frac{\cosh^2 x - \sinh^2 x}{\cosh^2 x} = \frac{\cosh^2 x}{\cosh^2 x} - \frac{\sinh^2 x}{\cosh^2 x}$$

$$\boldsymbol{\frac{d\left(\frac{\sinh x}{\cosh x}\right)}{dx} = 1 - \tanh^2 x}$$

ReLu:

$$f(x) = max(0,x)$$

$$f(x) = \begin{cases} x, & x > 0 \\ 0, & \text{otherwise} \end{cases}$$

$$f'(x) = \begin{cases} 1, & x > 0 \\ 0, & \text{otherwise} \end{cases}$$

1c) Building the Neural Network

Three Layer Network

$$z^1 = W^1 x + b^1 \tag{1}$$
$$a^1 = actFun(z^1) \tag{2}$$
$$z^2 = W^2 a^1 + b^2 \tag{3}$$
$$a^2 = \hat{y} = softmax(z^2) \tag{4}$$

Mean Cross Entropy Loss of Batch

$$L(y, \hat{y}) = -\frac{1}{N} \sum_{n=1}^{N} \sum_{i=1}^{C} y_{n,i} \log \hat{y}_{n,i} \tag{5}$$

1d) Backward Pass - Backpropagation

Gradients: $\frac{\partial L}{\partial W^2}, \frac{\partial L}{\partial b^2}, \frac{\partial L}{\partial W^1}, \frac{\partial L}{\partial b^1}$

$$\frac{\partial L}{\partial W^2} = \frac{\partial L}{\partial \hat{y}} \frac{\partial \hat{y}}{\partial z^2} \frac{\partial z^2}{\partial W^2}$$

$$\frac{\partial z^2}{\partial W^2} = \frac{\partial (W^2 a^2 + b^2)}{\partial W^2}$$

$$\boldsymbol{\frac{\partial z^2}{\partial W^2} = a^2}$$

$$\frac{\partial \hat{y}_i}{\partial z_i^2} = \frac{\partial \left(\frac{e^{z_i^2}}{\sum_{j=1}^{C} e^{z_j^2}} \right)}{\partial z_i^2}$$

if $j = i$

$$= \frac{e^{z_i^2} \sum_{j=1}^{C} e^{z_j^2} - e^{z_i^2} e^{z_j^2}}{\left(\sum_{j=1}^{C} e^{z_j^2} \right)^2} = \frac{e^{z_i^2} (\sum_{j=1}^{C} e^{z_j^2} - e^{z_j^2})}{\sum_{j=1}^{C} e^{z_j^2} \sum_{j=1}^{C} e^{z_j^2}}$$

$$\boldsymbol{\frac{\partial \hat{y}_i}{\partial z_i^2} = \hat{y}_i (1 - \hat{y}_i)}$$

if $j \neq i$

$$\frac{\partial \hat{y}_i}{\partial z_j^2} = \frac{0 - e^{z_i^2} e^{z_j^2}}{(\sum_{j=1}^{C} e^{z_j^2})^2}$$

$$\boldsymbol{\frac{\partial \hat{y}_i}{\partial z_j^2} = -\hat{y}_i \hat{y}_j}$$

$$\frac{\partial L}{\partial \hat{y}_i} = -\frac{1}{N} \sum_{n=1}^{N} \sum_{i=1}^{C} \frac{\partial (y_i \log \hat{y}_i)}{\partial \hat{y}_i} = -\frac{1}{N} \sum_{n=1}^{N} \sum_{i=1}^{C} y_i \left(\frac{1}{\hat{y}_i}\right) \frac{\partial \hat{y}_i}{\partial z_i^2}$$

$\forall i$

$$= -\frac{1}{N} \sum_{n=1}^{N} \left[\frac{y_i}{\hat{y}_i} (1 - \hat{y}_i) + \sum_{j \neq i}^{C} \frac{y_j}{\hat{y}_j} (-\hat{y}_j \hat{y}_i) \right]$$

$$= -\frac{1}{N} \sum_{n=1}^{N} \left[y_i (1 - \hat{y}_i) - \sum_{j \neq i}^{C} y_j \hat{y}_i \right]$$

$$= \frac{1}{N} \sum_{n=1}^{N} \left[-y_i + y_i \hat{y}_i + \sum_{j \neq i}^{C} y_j \hat{y}_i \right]$$
$$= \frac{1}{N} \sum_{n=1}^{N} \left[-y_i + \sum_{j=1}^{C} y_j \hat{y}_i \right]$$

$= \text{since } \boldsymbol{y_j} \text{ is one-hot encoded}$

$$\implies \sum_{j=1}^{C} y_j = 1$$

$$\therefore \boldsymbol{\frac{\partial L}{\partial z_i^2}} = \boldsymbol{\frac{1}{N} \sum_{n=1}^{N} (\hat{y}_i - y_i)}$$

Let $\delta^3 = (\hat{y}_i - y_i)$

$$\frac{\partial L}{\partial W^2} = \frac{1}{N} \sum_{n=1}^{N} \delta^3 a^3$$

$$\boldsymbol{\frac{\partial L}{\partial W^2} = \frac{1}{N} a^{1T} \delta^3}$$

$$\frac{\partial L}{\partial b^2} = \frac{\partial L}{\partial \hat{y}}\frac{\partial \hat{y}}{\partial z^2}\frac{\partial z^2}{\partial b^2}$$

$$\frac{\partial z^2}{\partial b^2} = 1$$

$$\boldsymbol{\frac{\partial L}{\partial b^2} = \frac{1}{N}\sum_{n=1}^{C}\delta^3}$$

$$\frac{\partial L}{\partial W^1} = \underbrace{\frac{\partial L}{\partial \hat{y}}\frac{\partial \hat{y}}{\partial z^2}}_{\frac{1}{N}\sum_{i=1}^{N}\delta^3}\overbrace{\frac{\partial z^2}{\partial a^1}}^{W^2}\underbrace{\frac{\partial a^1}{\partial z^1}}_{f'(z^1)}\overbrace{\frac{\partial z^1}{\partial W^1}}^{x}$$

$$= \frac{1}{N}W^{2T}\delta^3 f'(z^1)$$

$$\textbf{Let } \delta^2 = W^{2T}\delta^3 f'(z^1)$$

$$= \frac{1}{N}\delta^2 x$$

$$\boldsymbol{\frac{\partial L}{\partial W^1} = \frac{1}{N}x^T\delta^2}$$

$$\frac{\partial L}{\partial b^1} = \underbrace{\frac{\partial L}{\partial \hat{y}}\frac{\partial \hat{y}}{\partial z^2}\frac{\partial z^2}{\partial a^1}\frac{\partial a^1}{\partial z^1}}_{\frac{1}{N}\delta^2}\overbrace{\frac{\partial z^1}{\partial b^1}}^{1}$$

$$\boldsymbol{\frac{\partial L}{\partial b^1} = \frac{1}{N}\delta^2}$$

1e) Training Network

1e.2Using more Hidden Neurons

Incresing the number of neurons makes it have more degrees of freedome.

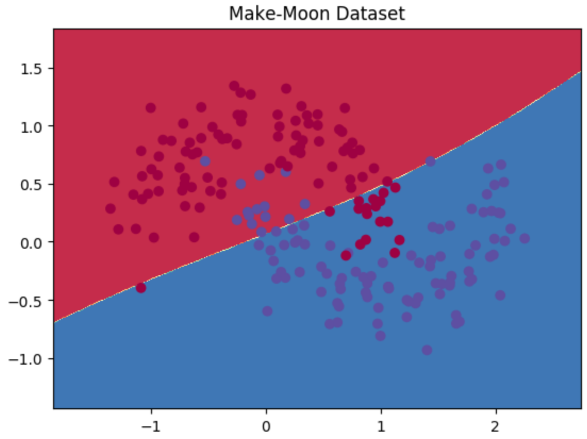

Figure 2: Trained Network Using Sigmoid Activation Function

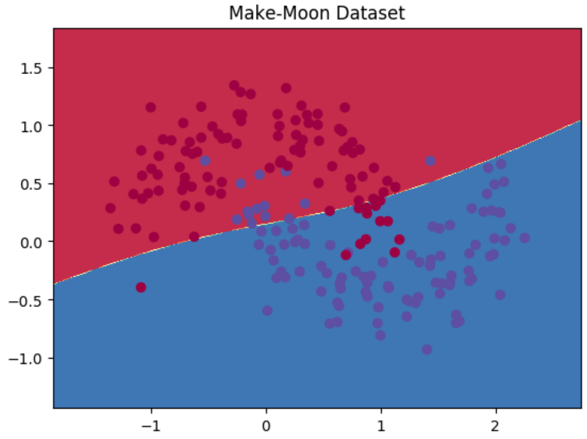

Figure 3: Trained Network Using Tanh Activation Function

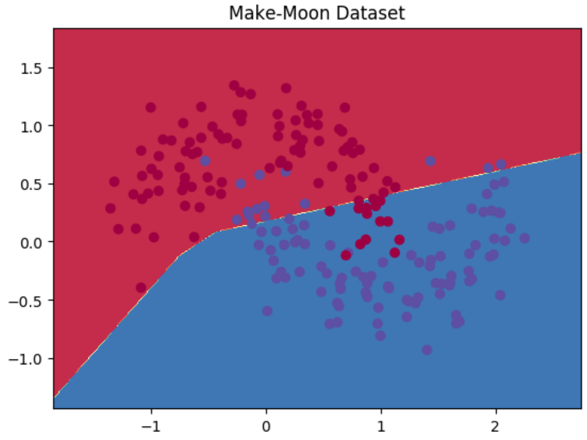

Figure 4: Trained Network Using ReLu Activation Function

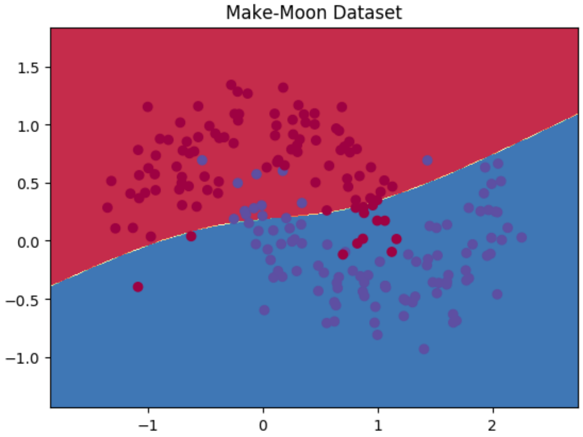

Figure 5: Trained Network Using Tanh Activation Function and 50 Hidden Neurons

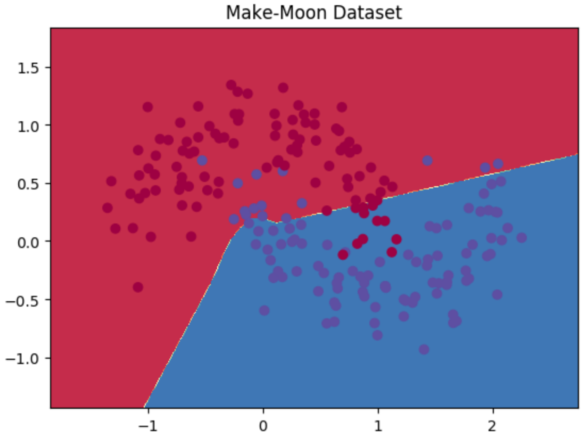

Figure 6: Trained Network Using ReLu Activation Function and 50 Hidden Neurons